

[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Fostling Counts](#)[Show S Numbers](#)[Edit S Numbers](#)[Preferences](#)[Cases](#)**Search Results -**

Terms	Documents
(dynamic ADJ instrumentation)	22

Database:

US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

L1

[Refine Search](#)[Recall Text](#)[Clear](#)**Search History****DATE:** Sunday, November 09, 2003[Printable Copy](#)[Create Case](#)**Set Name Query**

side by side

Hit Count Set Name

result set

DB=USPT; PLUR=NO; OP=OR

L1 (dynamic ADJ instrumentation)

22

L1

END OF SEARCH HISTORY

[Generate Collection](#)[Print](#)**Search Results - Record(s) 1 through 22 of 22 returned.**☐ 1. Document ID: US 6587967 B1

L1: Entry 1 of 22

File: USPT

Jul 1, 2003

US-PAT-NO: 6587967

DOCUMENT-IDENTIFIER: US 6587967 B1

TITLE: Debugger thread monitor

DATE-ISSUED: July 1, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bates; Cary Lee	Rochester	MN		
Ryan; Jeffrey Michael	Byron	MN		
Day; Paul Reuben	Rochester	MN		

US-CL-CURRENT: 714/35; 714/38

ABSTRACT:

A computer system, program product and method that monitor the threads executing within a region of a computer program during debugging. This region may be a plurality of nonadjacent sections of code, each with defined entry and exit addresses identified by control points. Some or all threads may be halted depending on a predetermined criteria related to threads of interest or the number of threads executing in the region. Of special interest is monitoring for a thread count so that timing errors may be analyzed for when some plurality of threads simultaneously execute within the region. Moreover, in the illustrative embodiment, control points are implemented for thread monitoring in a manner similar to a break point, utilizing a break point table to determine whether a system exception is due to a break point or to a thread monitor control point. If the latter, program execution continues after storing the thread identifier in a record for the thread monitor control point in the break point table.

20 Claims, 8 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 6

Full	Title	Class	Front	Review	Classification	Date	Reference	Sequences	Attachments	Claims	PMO	Draw Desc	Image
------	-------	-------	-------	--------	----------------	------	-----------	-----------	-------------	--------	-----	-----------	-------

☐ 2. Document ID: US 6546553 B1

L1: Entry 2 of 22

File: USPT

Apr 8, 2003

US-PAT-NO: 6546553

DOCUMENT-IDENTIFIER: US 6546553 B1

TITLE: Service installation on a base function and provision of a pass function with a service-free base function semantic

DATE-ISSUED: April 8, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Hunt; Galen C.	Bellevue	WA		

US-CL-CURRENT: 717/174; 712/233, 712/234, 712/244, 714/38, 717/163, 717/175, 717/176, 717/177

ABSTRACT:

A base function provides a base function semantic. During service installation, an unconditional branch instruction to a service function replaces one or more instructions at the beginning of a base function. The service function provides a service semantic such as instrumentation, redirection, replacement, or extension. After service installation, a pass function includes the replaced base function instructions and an unconditional branch instruction to the logically subsequent base function instruction. Thus, the pass function provides a service-free base function semantic. The service function calls the pass function an arbitrary number of times before and/or after executing any other service function instructions. The pass function is allocated statically or dynamically. A statically allocated pass function is callable before and/or after service installation to guarantee a service-free base function semantic. A service removal function restores a base function and conforms a pass function to the restored base function. A pass function is callable before and/or after service removal. A library of service installation functions includes functions for installing and removing a service on a base function. A library of binary editing functions includes functions for attaching service installation functions and associated data payloads to a binary file.

57 Claims, 9 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 9

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments	Claims	WAC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------	--------	-----	-----------	-------

☐ 3. Document ID: US 6545549 B2

L1: Entry 3 of 22

File: USPT

Apr 8, 2003

US-PAT-NO: 6545549

DOCUMENT-IDENTIFIER: US 6545549 B2

TITLE: Remotely controllable phase locked loop clock circuit

DATE-ISSUED: April 8, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Swoboda; Gary L.	Sugarland	TX		

US-CL-CURRENT: 331/18; 327/156, 327/159, 331/1A, 331/16, 331/25, 331/57

ABSTRACT:

This invention is a remotely controllable clock circuit embodied in a single integrated circuit device. The clock circuit includes at least one externally writable clock control register, a reference clock input, a controllable oscillator circuit, a pre-scalar circuit and a comparison circuit. The comparison circuit controlling the frequency of the controllable oscillator circuit to achieve a frequency match between a pre-scaled reference clock signal and a pre-scaled oscillator clock signal. The pre-scale divide factors are stored in respective fields in the clock control register. The clock control register may be memory mapped into a device memory space, accessed via an indirect access register or accessed via a serial scan chain. A trace first-in-first-out buffer has an input for

trace data operating under the function clock signal of the operating circuits and an output operating under the oscillator clock signal.

5 Claims, 13 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 8

Full Title Edition Front Review Classification Date Reference Sequences Attachments

PMAC Draw Desc Image

☐ 4. Document ID: US 6530079 B1

L1: Entry 4 of 22

File: USPT

Mar 4, 2003

US-PAT-NO: 6530079
DOCUMENT-IDENTIFIER: US 6530079 B1

TITLE: Method for optimizing locks in computer programs

DATE-ISSUED: March 4, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Choi; Jong-Deok	Mount Kisco	NY		
Gupta; Manish	Peekskill	NY		
Serrano; Mauricio J.	San Jose	CA		
Sreedhar; Vugranam C.	Whiteplains	NY		
Midkiff; Samuel Pratt	Upper Saddle River	NJ		

US-CL-CURRENT: 717/158; 712/227, 717/127, 717/128, 717/129, 717/130, 717/131, 717/152, 717/153, 717/157, 717/159

ABSTRACT:

A method and several variants for using information about the scope of access of objects acted upon by mutual exclusion, or mutex, locks to transform a computer program by eliminating locking operations from the program or simplifying the locking operations, while strictly performing the semantics of the original program. In particular, if it can be determined by a compiler that the object locked can only be accessed by a single thread it is not necessary to perform the "acquire" or "release" part of the locking operation, and only its side effects must be performed. Likewise, if it can be determined that the side effects of a locking operation acting on a variable which is locked in multiple threads are not needed, then only the locking operation, and not the side effects, needs to be performed. This simplifies the locking operation, and leads to faster programs which use fewer computer processor resources to execute; and programs which perform fewer shared memory accesses, which in turn not only causes the optimized program, but also other programs executing on the same computing machine to execute faster. The method also describes how information about the semantics of the locking operation side effects and the information about the scope of access can also be used to eliminate performing the side effect parts of the locking operation, thereby completely eliminating the locking operation. The method also describes how to analyze the program to compute the necessary information about the scope of access. Variants of the method show how one or several of the features of the method may be performed.

21 Claims, 2 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 2

Full Title Edition Front Review Classification Date Reference Sequences Attachments

PMAC Draw Desc Image

☐ 5. Document ID: US 6484819 B1

L1: Entry 5 of 22

File: USPT

Nov 26, 2002

US-PAT-NO: 6484819

DOCUMENT-IDENTIFIER: US 6484819 B1

TITLE: Directional borehole drilling system and method

DATE-ISSUED: November 26, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Harrison; William H.	West Hills	CA	91307	

US-CL-CURRENT: 175/61; 166/66, 175/26, 175/279, 175/45, 175/73

ABSTRACT:

A directional borehole drilling system employs a controllable drill bit, which includes one or more drilling surfaces which are dynamically positionable in response to respective command signals. Instrumentation located near the bit measures present position when the bit is static, dynamic and drilling surface position information when the bit is rotating, and stores a desired trajectory. This data is processed to determine the error between the present position and the desired trajectory, and the position of one or more of the bit's drilling surfaces is automatically changed as needed to make the bit dig in the direction necessary to reduce the error. The controllable drill bit preferably comprises three cone assemblies mounted about the bit's central axis, each of which includes a cone and an eccentric cam that rotate about a common axle. In response to a command signal, the cam is locked to the cone to cause concentric rotation of the cone, or locked to the axle to cause eccentric rotation of the cone--which causes the bit to dig in a preferred direction.

29 Claims, 11 Drawing figures

Exemplary Claim Number: 23

Number of Drawing Sheets: 9

Full Title Citation Front Review Classification Date Reference Sequences Attachments

RMIC Draw Desc Image

☐ 6. Document ID: US 6430741 B1

L1: Entry 6 of 22

File: USPT

Aug 6, 2002

US-PAT-NO: 6430741

DOCUMENT-IDENTIFIER: US 6430741 B1

TITLE: System and method for data coverage analysis of a computer program

DATE-ISSUED: August 6, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mattson, Jr.; James S.	Campbell	CA		
Man; Richard F.	Palo Alto	CA		

US-CL-CURRENT: 717/154; 717/126, 717/158

ABSTRACT:

The inventive system and method is directed toward verifying the accuracy of data tables specified by a developer to be used by a program. The system searches through

an application program for instructions which access areas of memory declared by the developer as being of interest and executes instrumentation code for these instructions. Input to the program is the source code of a user program and optionally, a data coverage specification prepared by a developer. Instrumentation can be implemented by inserting instrumenting code into the source code prior to compilation using facilities within the compiler itself. Alternatively, the instrumentation code can be added to the executable program code after compilation is complete. Yet a third option involves generating and executing instrumentation during execution of the user program without ever modifying the user program code at any stage. The output of the system is data coverage information indicating the number of times that various elements of the data tables of interest have been accessed during one full run of the user program. The system thereby provides a mechanism for evaluating the integrity of data to be accessed by a program where prior instrumentation systems have concentrated on verifying the validity of program logic flow.

18 Claims, 11 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 5

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

FIG. 1	Draw Desc	Image
--------	-----------	-------

☒ 7. Document ID: US 6401240 B1

L1: Entry 7 of 22

File: USPT

Jun 4, 2002

US-PAT-NO: 6401240
DOCUMENT-IDENTIFIER: US 6401240 B1

TITLE: System and method for profiling code on symmetric multiprocessor architectures

DATE-ISSUED: June 4, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Summers; Chuck	Plano	TX		

US-CL-CURRENT: 717/130; 709/100

ABSTRACT:

A profiler that accurately measures performance metrics for all threads executing a process on a SMP computer system. The profiler uses dynamic instrumentation to cause threads to sample performance metrics before and after certain code regions. In addition, the profiler uses extensions to a parallel support layer to register a parent thread with its child threads. Each thread stores the measured performance metric, or delta, in a memory cell or cells corresponding to its region and its parent region. When the process is complete, the profiler scans through the memory storage areas and sums the deltas for each particular level of code. Then, the results may be analyzed at the thread or process level. In this manner, the profiler can be adapted to work with any process executing on the computer system.

20 Claims, 6 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 5

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

FIG. 1	Draw Desc	Image
--------	-----------	-------

☐ 8. Document ID: US 6388533 B1

US-PAT-NO: 6388533

DOCUMENT-IDENTIFIER: US 6388533 B1

TITLE: Programmable ring oscillator

DATE-ISSUED: May 14, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Swoboda; Gary L.	Sugarland	TX		

US-CL-CURRENT: 331/57; 331/1A, 331/17, 331/179

ABSTRACT:

A controllable ring oscillator clock circuit includes a plurality of ring oscillator stages disposed in a linear chain. Each stage has a latch that determines if this stage is the last stage in the ring. In a propagate state of the latch the ring pulse is sent to the next stage. In a return state of the latch the ring pulse is returned to the prior stage. The latches are programmed like a shift register. A more command transfers the propagate state to the next stage. This increases the length of the delay line and thus decreases the oscillator frequency. A less command transfers the return state to the prior stage, decreasing the ring delay and increasing the oscillator frequency. In the preferred embodiment the delay stages are deployed as even and odd pairs with only the even or the odd stages changed at one time. This enables a simple structure because the pairs operate like a master-slave flip-flop, that is the data can move only a single stage at a time.

11 Claims, 13 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 8

Full	Title	Publication	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	-------------	-------	--------	----------------	------	-----------	-----------	-------------

Table	Draw Desc	Image
-------	-----------	-------

☐ 9. Document ID: US 6378125 B1

L1: Entry 9 of 22

File: USPT

Apr 23, 2002

US-PAT-NO: 6378125

DOCUMENT-IDENTIFIER: US 6378125 B1

**** See image for Certificate of Correction ****

TITLE: Debugger thread identification points

DATE-ISSUED: April 23, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bates; Cary Lee	Rochester	MN		
Ryan; Jeffrey Michael	Byron	MN		

US-CL-CURRENT: 717/129; 717/138

ABSTRACT:

A computer system, program product and method to facilitate debugging of multi-threaded computer programs by identifying threads that interact with program elements such as instructions or variables. For instance, after program execution is halted by a break point, a user may request retrieval of an identification of all threads encountering a certain statement number. Moreover, the program element may

be implemented as a thread identification control point in a manner similar to a break point, utilizing a break point table to determine whether a system exception is due to a break point or to a thread identification control point. If the latter, program execution continues after storing the thread identifier in a record for the thread identification control point in the break point table.

21 Claims, 15 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 9

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

MMAC	Draw Desc	Image
------	-----------	-------

☐ 10. Document ID: US 6378124 B1

L1: Entry 10 of 22

File: USPT

Apr 23, 2002

US-PAT-NO: 6378124
DOCUMENT-IDENTIFIER: US 6378124 B1

TITLE: Debugger thread synchronization control points

DATE-ISSUED: April 23, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bates; Cary Lee	Rochester	MN		
Ryan; Jeffrey Michael	Byron	MN		

US-CL-CURRENT: 717/129; 714/707, 717/138

ABSTRACT:

A computer system, program product and method utilize thread synchronization for debugging multi-threaded computer programs. Synchronization control points ("sync points") are used to conditionally suspend or delay execution of a thread or threads depending on another thread or threads hitting the same or other sync points. A thread hitting a synchronization control point is suspended, reference is made to break point table to determine what synchronization condition must be triggered to release the thread, and if triggered, what delay, if any, will be imposed prior to release.

17 Claims, 10 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 6

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

MMAC	Draw Desc	Image
------	-----------	-------

☐ 11. Document ID: US 6370457 B1

L1: Entry 11 of 22

File: USPT

Apr 9, 2002

US-PAT-NO: 6370457
DOCUMENT-IDENTIFIER: US 6370457 B1

TITLE: Debriefing systems and methods for retrieving and presenting multiple datastreams with time indication marks in time synchronism

DATE-ISSUED: April 9, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Nemeth; Louis G.	Lake Wylie	SC		

US-CL-CURRENT: 701/35; 434/35, 701/14, 701/3

ABSTRACT:

Event debriefing systems and methods are provided for recording a plurality of datastreams related to an event. An observer of the event may create time indication marks that facilitate recalling, during debriefing, what transpired at any particular point in time during the event. The recorded datastreams may be retrieved and presented beginning at marked time indications and continued chronologically thereafter in time synchronism.

36 Claims, 11 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 8

Full	Title	Edition	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	---------	-------	--------	----------------	------	-----------	-----------	-------------

SMC	Draw Desc	Image
-----	-----------	-------

☒ 12. Document ID: US 6216237 B1

L1: Entry 12 of 22

File: USPT

Apr 10, 2001

US-PAT-NO: 6216237

DOCUMENT-IDENTIFIER: US 6216237 B1

TITLE: Distributed indirect software instrumentation

DATE-ISSUED: April 10, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Klemm; Reinhard	North Plainfield	NJ		
Singh; Navjot	Morristown	NJ		
Tsai; Timothy	North Plainfield	NJ		

US-CL-CURRENT: 714/38; 714/35

ABSTRACT:

The invention provides a software instrumentation tool operative to control the execution of a target program, and to execute user-specified instrumentation actions upon occurrence of corresponding user-specified events during target program execution. Advantageously, the instrumentation tool permits the instrumentation actions to be implemented without modification of the target program code, and can be used to provide any desired type of instrumentation on any target program. In an illustrative embodiment, the instrumentation tool includes a frontend portion which provides a creation graphical user interface (GUI) to the tool, and a backend portion which controls execution of the target program and executes the user-specified actions. The frontend portion also includes, for a given target program, a created GUI which is created by a user for providing selected instrumentation functions for the given target program. The frontend and backend portions of the instrumentation tool may each be running on a different machine, or both may run on the same machine. The invention may be used in a wide variety of applications, including application program modification and management, fault tolerance, real-time collaboration, process monitoring, software rejuvenation and graphical interface generation.

33 Claims, 9 Drawing figures
Exemplary Claim Number: 1

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

FORM	Draw Desc	Image
------	-----------	-------

☐ 13. Document ID: US 6164841 A

L1: Entry 13 of 22

File: USPT

Dec 26, 2000

US-PAT-NO: 6164841

DOCUMENT-IDENTIFIER: US 6164841 A

TITLE: Method, apparatus, and product for dynamic software code translation system

DATE-ISSUED: December 26, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mattson, Jr.; James S.	Campbell	CA		
Shah; Lacky V.	Sunnyvale	CA		
Buzbee; William B.	Half Moon Bay	CA		
Benitez; Manuel E.	Cupertino	CA		

US-CL-CURRENT: 716/1

ABSTRACT:

A method and apparatus for improving the process of software development by a dynamic software development tool. The present invention efficiently executes in a user process and provides software developers with a high performance tool for software optimization. The present invention may augment the user process code instructions at runtime and, for every series of machine instructions that the original user source code would have executed, a series of instructions may be executed that are semantically equivalent to the user process code instructions and are altered to optimize the user process code instructions. The present invention may use emulation or translation to alter the user process code instructions. The resulting process is executed in the user process space and advantageously maintains the original flow of instruction execution. The present invention employs a technique of dynamically translating code at runtime and may operate on a virtual machine or a hardware machine. Further, the present invention may operate on various types of code segments and is not, for instance, limited to manipulating only code modules and may operate on a basic code block. Therefore, the present invention enables efficient optimized user process code instruction debugging, instrumentation, and simulation by preserving the original flow of user process code instructions while enabling emulation and translation on the original user process code instructions.

25 Claims, 7 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 5

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

FORM	Draw Desc	Image
------	-----------	-------

☐ 14. Document ID: US 5911073 A

L1: Entry 14 of 22

File: USPT

Jun 8, 1999

US-PAT-NO: 5911073

DOCUMENT-IDENTIFIER: US 5911073 A

**** See image for Certificate of Correction ****

TITLE: Method and apparatus for dynamic process monitoring through an ancillary control code system

DATE-ISSUED: June 8, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mattson, Jr.; James S.	Campbell	CA		
Shah; Lacky V.	Sunnyvale	CA		
Buzbee; William B.	Half Moon Bay	CA		

US-CL-CURRENT: 717/104

ABSTRACT:

A method and apparatus for improving the process of software development by a dynamic software development tool. The present invention allows the execution of an emulation tool to occur under the control of the original user process and preserves the execution flow of the user process instructions. The present invention manages the execution of the emulation tool within the computer memory. The present invention uses the user process code as data to direct the execution of the emulation tool. The present invention enables the use of other software development tools such as monitoring and profiling tools, program analysis tools, simulation tools, and software debugging tools.

18 Claims, 11 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 7

Full Title Citation Front Review Classification Date Reference Sequences Attachments

SMC Draw Desc Image

☒ 15. Document ID: US 5909578 A

L1: Entry 15 of 22

File: USPT

Jun 1, 1999

US-PAT-NO: 5909578

DOCUMENT-IDENTIFIER: US 5909578 A

TITLE: Use of dynamic translation to burst profile computer applications

DATE-ISSUED: June 1, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Buzbee; William B.	Half Moon Bay	CA		

US-CL-CURRENT: 717/130; 717/131, 717/136

ABSTRACT:

A method and system for burst profiling an application program. The native application executes free and unfettered for a first time period. Then, the native application is halted and is instrumented by a dynamic translator. The dynamic translator translates and instruments code blocks of the application as the application is executing. The instrumented application executes and gathers profile data for a second time period. When the second time period expires, the dynamic translator continues to execute the instrumented application until it reaches a known state. Then, the instrumented application is halted at that known state. Next, the native application resumes execution from the known state. This process repeats until either the application finishes execution or enough profile data has been collected.

18 Claims, 4 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 3

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

WWW	Draw Desc	Image
-----	-----------	-------

☐ 16. Document ID: US 5872909 A

L1: Entry 16 of 22

File: USPT

Feb 16, 1999

US-PAT-NO: 5872909

DOCUMENT-IDENTIFIER: US 5872909 A

TITLE: Logic analyzer for software

DATE-ISSUED: February 16, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Wilner; David N.	Oakland	CA		
Smith; Colin	Alameda	CA		
Cohen; Robert D.	Oakland	CA		
Burd; Dana	Oakland	CA		
Fogelin; John C.	Berkeley	CA		
Fox; Mark A.	San Francisco	CA		
Long; Kent D.	Richmond	CA		
Burns; Stella M.	San Francisco	CA		

US-CL-CURRENT: 714/38; 714/47

ABSTRACT:

The present invention logs events which occur in the target software, and stores these in a buffer for periodic uploading to a host computer. Such events include the context switching of particular software tasks, and task status at such context switch times, along with events triggering such a context switch, or other events. The host computer reconstructs the real-time status of the target software from the limited event data uploaded to it. The status information is then displayed in a user-friendly manner. This provides the ability to perform a logic analyzer function on real-time software. A display having multiple rows, with one for each task or interrupt level, is provided. Along a time line, an indicator shows the status of each program, with icons indicating events and any change in status.

30 Claims, 18 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 14

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

WWW	Draw Desc	Image
-----	-----------	-------

☒ 17. Document ID: US 5838976 A

L1: Entry 17 of 22

File: USPT

Nov 17, 1998

US-PAT-NO: 5838976

DOCUMENT-IDENTIFIER: US 5838976 A

TITLE: System and method for profiling code on symmetric multiprocessor

architectures

DATE-ISSUED: November 17, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Summers; Chuck	Plano	TX		

US-CL-CURRENT: 717/130; 717/119

ABSTRACT:

A profiler that accurately measures performance metrics for all threads executing a process on a SMP computer system. The profiler uses dynamic instrumentation to cause threads to sample performance metrics before and after certain code regions. In addition, the profiler uses extensions to a parallel support layer to register a parent thread with its child threads. Each thread stores the measured performance metric, or delta, in a memory cell or cells corresponding to its region and its parent region. When the process is complete, the profiler scans through the memory storage areas and sums the deltas for each particular level of code. Then, the results may be analyzed at the thread or process level. In this manner, the profiler can be adapted to work with any process executing on the computer system.

37 Claims, 6 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 5

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------

FIGURE	Drawing Description	Image
--------	---------------------	-------

☐ 18. Document ID: US 5826206 A

L1: Entry 18 of 22

File: USPT

Oct 20, 1998

US-PAT-NO: 5826206

DOCUMENT-IDENTIFIER: US 5826206 A

**** See image for Certificate of Correction ****

TITLE: Debriefing systems and methods for retrieving and presenting multiple datastreams with time indication marks in time synchronism

DATE-ISSUED: October 20, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Nemeth; Louis G.	Lake Wylie	SC		

US-CL-CURRENT: 701/35; 360/27, 360/72.2, 386/69, 386/70, 701/14, 701/3

ABSTRACT:

Event debriefing systems and methods are provided for recording a plurality of datastreams related to an event. Each datastream includes chronological information related to at least one aspect of the event. An observer of the event may create time indication marks that facilitate recalling, during debriefing, what transpired at any particular point in time during the event. The recorded datastreams may be retrieved and presented beginning at marked time indications and continued chronologically thereafter in time synchronism.

30 Claims, 9 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 7

☒ 19. Document ID: US 5313616 A

L1: Entry 19 of 22

File: USPT

May 17, 1994

US-PAT-NO: 5313616

DOCUMENT-IDENTIFIER: US 5313616 A

**** See image for Certificate of Correction ****

TITLE: Method for analyzing calls of application program by inserting monitoring routines into the executable version and redirecting calls to the monitoring routines

DATE-ISSUED: May 17, 1994

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Cline; David C.	San Jose	CA		
Silverman; Andrew P.	Los Gatos	CA		
Wymore; Farrell W.	Mountain View	CA		

US-CL-CURRENT: 717/127; 713/323, 714/35, 714/45, 717/130

ABSTRACT:

A method for verifying the conformance of an application program to a set of system rules characterized by the development of a conformance database, the performance of a static analysis of the application program to determine whether the application program is in static conformance with the conformance database and the performance of a dynamic analysis of the application program to determine whether the application program is in dynamic conformance with the conformance database. The static analysis produces a graph of the basic blocks of the application program and analyzes the graph for conformance to system rules, dead code and coverage metrics. The dynamic analysis adds a small amount of monitoring code into an executable application program which monitors the application program as it is exercised in a test harness. The monitoring code produces a log database which can be analyzed for run-time non-conformities of the application program.

11 Claims, 17 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 15

☐ 20. Document ID: US 5088859 A

L1: Entry 20 of 22

File: USPT

Feb 18, 1992

US-PAT-NO: 5088859

DOCUMENT-IDENTIFIER: US 5088859 A

TITLE: Riser and tendon management system

DATE-ISSUED: February 18, 1992

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Devlin; Paul V.	Pearland	TX		

US-CL-CURRENT: 405/211; 405/224

ABSTRACT:

A riser and tendon management system for offshore hydrocarbon production facilities has a data acquisition system with instrumentation suitable for gathering all necessary information concerning the immediate condition of the facility and a riser and tendon analysis system for comparing actual with ideal conditions for the facility. Information is generated as to what corrective action must be taken. The analysis system stores past corrective actions and results and factors this information into the current suggestion for correction thereby reducing riser and tendon stress while increasing their fatigue life.

13 Claims, 3 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 3

Full Title Citation Front Review Classification Date Reference Sequences Attachments

RWC Draw Desc Image

☐ 21. Document ID: US 4934933 A

L1: Entry 21 of 22

File: USPT

Jun 19, 1990

US-PAT-NO: 4934933

DOCUMENT-IDENTIFIER: US 4934933 A

TITLE: Dental work station

DATE-ISSUED: June 19, 1990

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Fuchs; Jack	Louisville	KY	40242	

US-CL-CURRENT: 433/79; 312/209

ABSTRACT:

A dental work station designed to position the dental assistant's instrumentation and the dentist's dynamic instrumentation in an efficient location to deliver dental instruments to the respective operators in the best position in relation to the patient for both the comfort of the patient and the ease of the operators is shown. Due to the unique design of the work station, the instrumentation of both the assistant and the operator can switch from a left-handed to right-handed and reverse itself in a matter of seconds, thus making it a completely ambidextrous unit. The unique arrangement of the overhead structure allows the dentist's dynamic instruments to be efficiently delivered to the dentist without interfering with the dental assistant's equipment. In addition, the overhead structure houses both the astral operating light and the general light fixtures to provide optimum lighting for the operators. The overhead structure also contains a plurality of pipe chases to deliver essential dental utilities from the utility junction box located in the base of the service cabinet to the dentist's instruments. As added equipment there may also be provided a television monitor, a video cassette recorder and a computer for use during the operation of the dentist. This work station with all of its features leaves the dentist's chair completely unencumbered with great allowances for any type of operating position, thus making it a completely self contained dental operating unit.

12 Claims, 10 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 8

☐ 22. Document ID: US 4783994 A

L1: Entry 22 of 22

File: USPT

Nov 15, 1988

US-PAT-NO: 4783994

DOCUMENT-IDENTIFIER: US 4783994 A

TITLE: Pressure measuring probe

DATE-ISSUED: November 15, 1988

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ashby, Jr.; George C.	Newport News	VA		

US-CL-CURRENT: 73/147; 73/861.65

ABSTRACT:

The invention is a probe (10) for measuring changes in pressure in a high locity fluid streams over and adjacent the surface of an object. The probe (10) is formed of an exterior housing (11) having a closed pressure chamber (28) in which a piezo-electric pressure transducer (32) is mounted. An open connector tube (26) having a probe tip (29) passes a portion of said fluid stream into the closed pressure chamber (28), any change of pressure therein requiring a "settling-time" to appear in the closed pressure chamber (28) that is inversely proportional to the cross-sectional area of the connector tube (26). A cooling chamber (47) formed around the pressure chamber (28) is connected to a source of cooling fluid by means of inlet and outlet tubes (44) and (46), respectively.

9 Claims, 3 Drawing figures

Exemplary Claim Number: 8

Number of Drawing Sheets: 1

Generate Collection

Print

Terms	Documents
(dynamic ADJ instrumentation)	22

Display Format:

REV

Change Format

Previous Page

Next Page